

This Page Is Inserted by IFW Operations
and is not a part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

As rescanning documents *will not* correct images,
please do not report the images to the
Image Problem Mailbox.

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平8-37535

(43) 公開日 平成8年 (1996) 2月6日

| (51) Int. Cl. ° | 識別記号 | 庁内整理番号 | F I | 技術表示箇所 |
|-----------------|------|---------|-----------------------|--------|
| H 0 4 L 12/28 | | | H 0 4 L 11/00 3 1 0 B | |
| 12/56 | | 9466-5K | 11/20 1 0 2 D | |

審査請求 有 請求項の数 11 O L (全 26 頁)

(21) 出願番号 特願平6-314404

(22) 出願日 平成6年 (1994) 12月19日

(31) 優先権主張番号 1 7 9 3 9 7

(32) 優先日 1994年1月10日

(33) 優先権主張国 米国 (U S)

(71) 出願人 390009531

インターナショナル・ビジネス・マシーンズ・コーポレーション

INTERNATIONAL BUSINESS MACHINES CORPORATION

アメリカ合衆国10504、ニューヨーク州
アーモンク (番地なし)

(72) 発明者 チャールズ・エドワード・パーキンス

アメリカ合衆国10562 ニューヨーク州オ
ッシニング グレンデル・ロード 50

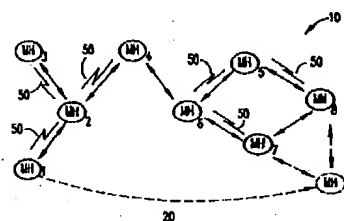
(74) 代理人 弁理士 合田 潔 (外2名)

(54) 【発明の名称】 経路指定方法

(57) 【要約】

【目的】 ネットワークの各局で記憶された経路指定テーブルを使って、移動局のアドホック・ネットワークの局間でパケットを伝送する。

【構成】 要求に応じて新しい経路を示し、最近変化した経路を周期的に示し、それにより頻度は低い全既知経路の完全ダンプを提供するという3つのスケジュールに従って、経路指定情報が同報通信またはマルチキャストされる。新しい経路により再同報通信が即座に開始され、その結果この情報が迅速に拡散される。経路は、周期的に公示される。この公示はほとんどの場合、主として、記憶されたすべてのものが、同報通信移動ホストに関してまだ正しいことをすべての隣接局に通知するために役立つ。振動を弱めるために、経路変化の頻度に関する情報が保持される。このデータに基づき、まもなく変化しようとする経路の公示を遅延するように決定が下され、これにより経路指定テーブルの振動を弱める。



【特許請求の範囲】

【請求項1】それぞれ固有ネットワーク・アドレスを持つが固定位置を持たない複数の移動ホストから構成されるアドホック・ネットワークに結合された2つの移動ホスト間で、パケットの情報を経路指定する方法であつて、

各移動ホストにおいて、ソース移動ホストから宛先移動ホストまでのホップ数として定義された「距離」を含む経路指定テーブルを記憶する段階と、

各移動ホストに記憶された経路指定テーブルを、その移動ホストが周期的に同報通信することにより経路を公示する段階と、

宛先移動ホストから発したタイムスタンプで、各経路指定テーブルのエントリをタグ付けする段階と、

他の移動ホストから受け取った同報通信に基づいて、各宛先移動ホストごとに、移動ホストに記憶された経路指定テーブルを更新する段階と、

各移動ホストが、隣接移動ホストから受け取った新しい経路指定情報を再伝送する段階と、

所望の宛先移動ホストに関する最良の「距離」を有する経路として、ソース移動ホストからパケットの情報を伝送するための経路を選択する段階とを含む方法。

【請求項2】アドホック・ネットワークが、ネットワーク層とリンク層とを含むネットワーク規格に従い、経路指定がアドホック・ネットワークのリンク層で実行されることを特徴とする、請求項1に記載の経路指定方法。

【請求項3】新しい経路が、より短い距離または無限の距離を有する経路と定義され、無限の距離は、破壊されたリンクを表し、すなわち特定の宛先が到達可能でなくなり、したがってこの新規の到達不可能な宛先に依存する他のすべての宛先がそれ自体到達不可能になることを意味し、隣接移動ホストから受け取った新しい経路指定情報を再伝送する前記段階が、新しい経路指定情報の受信時に移動ホストによって即時実行されることを特徴とする、請求項1に記載の経路指定方法。

【請求項4】前記経路指定テーブルに記憶された経路が変化する頻度に関するデータを保持する段階と、

前記経路指定テーブルに記憶された経路の平均安定待ち時間を決定することによって、経路の安定性を測定する段階と、

測定された経路の安定待ち時間を安定待ち時間テーブルに、記憶する段階と、

公示段階の前に安定待ち時間テーブルにアクセスし、まもなく変化する可能性のある経路の公示を遅らせ、それにより前記経路指定テーブル内の情報の変動を少なくする段階とをさらに含む、請求項1に記載の経路指定方法。

【請求項5】古い測定値よりも大きな加重因子を有する特定の経路の安定待ち時間の最新の測定値をカウントすることにより、測定された経路安定待ち時間に重み付け

を行う段階をさらに含む、請求項4に記載の経路指定方法。

【請求項6】アドホック・ネットワークがさらに、移動ホストのネットワーク層アドレスに基づいて、前記経路指定テーブルに経路指定情報を記憶する段階と、前記経路指定テーブルに記憶された経路が変化する頻度に関するデータを保持する段階と、

前記経路指定テーブルに記憶された経路の平均安定待ち時間を決定することにより、経路の安定性を測定する段階と、

測定された経路の安定待ち時間を安定待ち時間テーブルに記憶する段階と、

公示段階の前に安定待ち時間テーブルにアクセスし、まもなく変化する可能性のある経路の公示を遅らせ、それにより前記経路指定テーブル内の情報の変動を少なくする段階とをさらに含む、請求項1に記載の経路指定方法。

【請求項7】移動ホストのリンク層アドレスに基づいて、前記経路指定テーブルに経路指定情報を記憶する段階と、

前記経路指定テーブルに記憶された経路が変化する頻度に関するデータを保持する段階と、

前記経路指定テーブルに記憶された経路の平均安定待ち時間を決定することにより、経路の安定性を測定する段階と、

測定された経路の安定待ち時間を安定待ち時間テーブルに記憶する段階と、

公示段階の前に安定待ち時間テーブルにアクセスし、まもなく変化する可能性のある経路の公示を遅らせ、それにより前記経路指定テーブル内の情報の変動を少なくする段階とをさらに含む、請求項1に記載の経路指定方法。

【請求項8】ネットワーク層プロトコル使用可能度データを宛先ごとに追跡する段階をさらに含む、請求項1に記載の経路指定方法。

【請求項9】それぞれ固有ネットワーク・アドレスを持つが固定位置をもたない複数の移動ホストから構成され、ネットワーク層とリンク層とを含み、ネットワーク規格に従うアドホック・ネットワークに結合された2つの移動ホスト間で、パケットの情報を経路指定する方法であつて、

各移動ホストにおいて、ソース移動ホストから宛先移動ホストまでのホップ数として定義された「距離」を含む経路指定テーブルを記憶する段階と、

各移動ホストに記憶された経路指定テーブルを、その移動ホストが周期的に同報通信することにより経路を公示する段階と、

宛先移動ホストから発したタイムスタンプで、各経路指定テーブルのエントリをタグ付けする段階と、

他の移動ホストから受け取った同報通信に基づき、移動

3

ホストに記憶された経路指定テーブルを移動ホストごとに更新する段階であって、前記更新が、より良い距離または無限の距離を有する経路と定義される新しい経路に限定され、無限の距離が破壊リンクを表し、すなわち、特定の宛先に到達可能でなくなり、したがってこの新規の到達不可能な宛先に依存する他のすべての宛先がそれ自体到達不可能になることを意味する更新段階と、隣接移動ホストから受け取った新しい経路指定情報を各移動ホストが再伝送する段階であって、移動ホストが新しい経路指定情報を受け取ると即時に実行される再伝送段階と、

所望の宛先移動ホストに関する最短の「距離」を有する経路として、ソース移動ホストからパケットの情報を伝送する経路を選択する段階とを含む方法。

【請求項10】前記経路指定テーブルに記憶された経路が変化する頻度に関するデータを保持する段階と、前記経路指定テーブルに記憶された経路の平均安定待ち時間を決定することにより、経路の安定性を測定する段階と、

測定した経路の安定待ち時間を安定待ち時間テーブルに記憶する段階と、

公示段階の前に安定待ち時間テーブルにアクセスし、まもなく変化する可能性のある経路の公示を遅らせ、それにより前記経路指定テーブル内の情報の変動を少なくする段階とを含む、請求項9に記載の経路指定方法。

【請求項11】古い事象よりも大きな加重因子を有する特定経路の安定待ち時間の最新の測定値をカウントすることによって、測定された経路安定待ち時間に重み付けを行う段階をさらに含む、請求項10に記載の経路指定方法。

【発明の詳細な説明】

【0001】

【産業上の利用分野】本発明は、一般に無線データ通信システムに関し、より詳細には移動コンピュータ用のリンク層経路指定に関する。

【0002】

【従来の技術】様々な多くのネットワーク・プロトコルが定義されている。たとえば、国際標準化機構（ISO）は、開放型システム間相互接続（OSI）アーキテクチャ用の規格を発表した国際団体である。国防データ・ネットワーク（DDN）規格は、ローカル・エリア・ネットワーク（LAN）の相互接続を支援するインターネット・プロトコル（IP）用の基準を確立している。IPは、ユーザに提供されるサービスを定義し、それらのサービスを支援するために必要な機構を指定している。この規格はまた、下位プロトコル層に必要なサ

4

ービスを定義し、上位と下位のインターフェースを記述し、実施に必要な実行環境サービスの概要を記述している。

【0003】伝送制御プロトコル（TCP）は、パケット交換コンピュータLANおよびインターネットワークにおいて、接続本位の信頼性の高い端末間データ伝送を提供トランスポート・プロトコルである。IPとTCPは、ネットワークまたはサブネットワークの境界を横切って接続するかまたは接続性を利用する可能性を有する、米国防総省（DoD）のすべてのパケット交換ネットワークを使用するために必須である。インターネットワーキングに使用されるそのようなネットワークにおいて、ホスト、フロント・エンド、ゲートウェイなどのネットワーク要素はTCP/IPを実施しなければならない。

【0004】IPは、パケット交換通信LANを相互接続してインターネットワークを構成するように設計されている。IPは、インターネット・ダイアグラムと呼ばれるデータのブロックをインターネットを介してソースから宛先まで伝送する。ソースおよび宛先は、同じサブネットワーク上または接続されたLAN上のいずれかにあるホストである。DDN規格は、ホストのIPを指定する。IPは、DoDアーキテクチャ・モデルで定義されているので、インターネットワーク層に存在する。したがって、IPは、トランスポート層のプロトコルにサービスを提供し、下位ネットワーク・プロトコルのサービスに依拠する。IPの下位に様々なネットワーク・アクセス・プロトコルが存在し、それには、たとえばイーサネット・プロトコル、X.25プロトコル、および本明細書において特に重要な無線媒体アクセス・プロトコルが含まれる。

【0005】インターネット・プロトコルは本来、それぞれ固有のインターネット・アドレスを指定されたユーザが、固定した位置でネットワークに接続されるという仮定のもとで開発された。しかし、無線プロトコルを使用する携帯用またはハンドヘルドのコンピュータに関しては、一般に、ネットワークのあちこちにユーザが移動するのが、例外というよりもむしろ通例である。その結果として、このタイプの使い方がインターネット・プロトコルの設計の暗黙的前提に反するという点で問題が生じる。

【0006】現在は、コンピュータ相互の位置に関して特別な前提を設けない限り、無線データ通信装置を有する移動コンピュータを、互いに接続を維持したまま自由に動き回れるようにする方法はない。ある移動コンピュータはしばしば、それら自体ではデータを直接に交換できない他の2台の移動コンピュータとデータを交換できることがある。その結果、会議室内のコンピュータ・ユーザは、特にユーザが室内をあちこち移動するとき、ネットワーク接続を維持するためにその仲間のどのコン

コンピュータを頼りにすることができるか予測できないことがある。

【0007】ホストに割り当てられたネットワーク層アドレスがネットワーク・トポロジに関して何の意味ももたないときに、最適ネットワーク層経路指定を移動ホストに提供することに関して問題が生じる。この問題が生じるのは、ホストが、ホストが移動する時でも固定したままの識別子を有し、同時にネットワーク層においてネットワーク層経路指定を実現可能にするのに十分な情報を提供する必要があるのである。

【0008】

【発明が解決しようとする課題】したがって、本発明の目的は、変化する任意の相互接続経路に沿って任意の時点でデータを交換できる移動コンピュータ群が、そのすべてのコンピュータに、データを交換できる（できれば複数ホップの）経路を与えることができる、データ通信システムを提供することである。

【0009】さらに、本発明のより具体的な目的は、固定基地局の助けなしでリンク層経路指定を使用することによって、常に変化する経路に沿って複数の移動コンピュータ間でデータが交換できる技法を提供することである。

【0010】

【課題を解決するための手段】本発明によれば、無線データ通信ネットワークの局間でパケットを経路指定するための方法および装置が提供される。ネットワークの各局に記憶された経路指定テーブルを使って、パケットがネットワークの局間で伝送される。各局で記憶された各経路指定テーブルは、その局からアクセス可能な各局のリストと、アクセス可能な各局に達するために必要なホップ数とを提供する。これらのテーブルを動的に変化するトポロジにおいて維持するために、リンク層パケットを各局から伝送してテーブルを更新する。これらのリンク層パケットは、各局からアクセス可能な局とこのアクセス可能な局に達するために必要なホップ数とを示す。

【0011】経路指定情報は、伝送されるリンク層パケットを、ネットワーク内での局の移動の結果検出されるトポロジの変化として周期的かつ増分的に同報通信またはマルチキャストすることによって公示される。また、変動を弱めるために、経路が変化する頻度に関する情報を保持する。このデータに基づいて、まもなく変化する経路の公示を遅らせそれによって経路指定テーブルの変動を弱める決定を下すことができる。干渉する2つのリンク層パケット間の変動を防ぐために、ある経路の公示が遅延される。

【0012】

【実施例】ここで図面、特に図1を参照すると、ネットワークのプロトコル層を例示する国防データ・ネットワーク（DDN）のアーキテクチャ図が示されている。最上層11は、様々な適用業務プロトコル111および適

用業務プログラム112と113を含むセッション層である。それに加えて通常、電子メール（eメール）適用業務プログラム114がある。これらは、適用業務固有のプロトコルによって、トランスポート層すなわち層4と通信する。たとえば、適用業務プロトコル111はその固有モード115を介して層4と通信し、適用業務プロトコル112はファイル転送プロトコル（FTP）116によって通信し、適用業務プログラムはテルネット・プロトコル117によって通信し、メール適用業務114は単純メール転送プロトコル（SMTP）118によって通信する。

【0013】層4すなわちトランスポート層は伝送制御プロトコル（TCP）トランスポート層12から構成してもよく、層3すなわちネットワーク層はインターネット・プロトコル（IP）層13から構成される。2つの層12と13は、共通の名前とアドレス空間を共用する一組のTCP/IPプロトコルを含む。

【0014】リンク層14すなわち層2は、リンク層と媒体アクセス制御（MAC）層から構成される。リンク層14には、イーサネット141、DDN規格X.25142、無線プロトコル143、およびトークンリング144を含む様々なネットワーク・アクセス・プロトコルが含まれる。これらは一般に、規格団体によって公表された規格によって定義されている。たとえば、イーサネットはIEEE（米国電気電子学会）規格802.3によって定義され、トークン・リングはIEEE規格802.5によって定義される。

【0015】最後に、層1（図示せず）は物理層である。この層は、データ符号化を含む配線、接続および伝送パラメータに関係する。インターネットワーキングの層およびプロトコルの詳細な情報は、E. コマー（Comer）著“Internetworking with TCP/IP, Volume I: Principles, Protocols, and Architecture”, Prentice-Hall (1990)を参照されたい。

【0016】従来の経路指定技法に従って、層3に同様の技法を使用してもよいが、本発明の好ましい実施例の実施は、図1に示したアーキテクチャのリンク層または層2として識別された部分を特に対象とする。本発明によれば、各移動局は、その現隣接局のそれぞれに、それ自体の隣接局のリストを公示する必要がある。このリスト内のエントリは経時的に動的にかなり変化し、したがって、すべての移動コンピュータが常にその群の他のすべての移動コンピュータを確実に探し出せるように十分頻繁に公示を行わなければならない。さらに、各移動コンピュータは、要求に応じて、データを別のコンピュータに中継することに同意する。このように、移動コンピュータは、データの目標が直接通信の範囲内でない場合でも、グループ内の他の移動コンピュータとデータを交換することができる。他のコンピュータを中間点として使用してデータを送るこの方法は、経路指定として知ら

れる。経路指定はこれまで、ネットワーク・プロトコルのリンク層では移動コンピュータ・システムに関する上記の問題には適用されていなかった。群中の特定のコンピュータから他のどの移動コンピュータがアクセス可能であるかという通知は、本発明によればプロトコルのリンク層で行われ、したがって上位層（たとえばネットワーク層）のプロトコルが使用中であっても本発明の方法は動作する。集合内の移動コンピュータ群は、事実上新しい「ネットワーク」を構成し、管理なしでそうする。この通信の形式は、「アドホック (ad-hoc)」ネットワークと呼ばれている。

【0017】移動コンピュータはしばしば、建物内の有線バックボーンに沿ってネットワーク接続を維持する他のコンピュータとデータを交換することを可能にする「基地局」と一緒に使用される。この場合、基地局は常に多量の電力を有すると想定され、それに対して移動コンピュータは電力供給が厳しく制限されていることがあるので、経路指定機能は主に基地局が引き受ける。基地局は、移動コンピュータが基地局の範囲内のあらゆる移動コンピュータにアクセスを持つことを各基地局に任せるような形でリンク層経路指定に関与する。移動コンピュータが基地局の範囲内にある場合は、その隣接局リストの周期的な同報通信によって、基地局に直接接続性を公示する。基地局は、異なるセルの移動コンピュータ間でデータ交換経路を作成するためにどの移動局が協力できるかというリストを同報通信することもしないこともある。

【0018】上記の経路指定機能はすべてリンク層アドレス指定技法（いわゆる「ハードウェア・アドレス指定」）を使用するので、上記の方法は単一の物理媒体を使用する導入設備にほとんど適用できる。異なるネットワークにあるコンピュータが、ネットワーク層技法を使用してデータ交換用の経路を確立し維持する。本明細書に記載するリンク層経路指定の方法は、そのような他の技法と一緒に使用することができ、あるいは層2（すなわち「ハードウェア」）のアドレスの代わりに層3のアドレスを考慮し、層3のプロトコル・リストの伝送を無視することによって、層3で使用するためにのみ適合させることもできる。リンク層で単一のネットワーク・データ経路を提示することができ、ネットワーク層で複数のネットワーク・データ経路を処理することができる。複数ネットワーク設備における基地局は、必要なデータ経路をリンク層とネットワーク層のどちらで確立するかを決定する責任がある。基地局を含まないときは、本明細書で述べるリンク層経路指定をもっぱら移動コンピュータが使用する。

【0019】それら自体の間でデータ経路を作成するために相互動作するコンピュータはすべて、周期的に、たとえば1秒に1回または数秒に1回（あるいは、結局、新たな同報通信が必要なことを確実に決定できる方法が

設計されているときは、必要とされるだけ）必要データを同報通信する。各移動コンピュータによるデータ同報通信は、各宛先について、少なくとも以下の情報を有する経路指定テーブルを含む。

- ・宛先のリンク層アドレス。
- ・宛先に達するのに必要なホップ (hop) 数。
- ・元々宛先によってスタンプされた、その宛先に関して受け取った情報のタイムスタンプ。
- ・宛先が省略ルータとして働くことを望むかどうか

10 (たとえば、宛先が基地局である)。また伝送された経路指定テーブルは、リンク層ソフトウェアの当然の動作によって、それらを伝送する移動コンピュータのハードウェア・アドレスを含む。経路指定テーブルはまた、送信側によって作成されたタイムスタンプも含む。さらに、特定の移動コンピュータが同一セル内でアクセス可能か否かを決定しようとする経路指定アルゴリズムでは、どの基地局が各移動コンピュータにサービスしているかを示す表示を含んでもよい。

【0020】そのような経路指定テーブルを移動コンピュータが受け取ると、そのコンピュータは、ローカルに記憶されたそれ自体の経路指定テーブルを更新し始める。各受信経路は、同じ宛先と次ホップを示す全ての既存の経路を更新する。タイムスタンプは新しいタイムスタンプで置き換えられ、各受信経路中に示されたホップ数は増分され記憶される。任意のコンピュータまでの経路に沿ったホップ数は、その経路に関する「距離 (metric)」と呼ばれる。経路指定テーブルの送信側のアドレスは、宛先までの進路に沿った次ホップのアドレスとして記憶される。(最後の) 宛先までの経路に沿ったさらに他のアドレスは記憶する必要がない。既存のいずれの(宛先、次ホップ) アドレス対にも対応しない経路エントリを受け取ったときは、新しいエントリがその宛先に割り振られる。伝送される経路指定テーブルを発したコンピュータに関してローカル経路指定テーブルのエントリが作成または更新され、そのエントリに関する距離は1になり、コンピュータに達するためにただ1回のホップを必要とすることを示す。言い換えると、この2つのコンピュータ（経路指定テーブルの送信側と受信側）は「隣接局」である。

40 【0021】各宛先は、その代替経路指定経路に沿った次ホップに関して、異なるリンク層アドレスでそれぞれ指定された、限られた（少ない）数の代替経路を有する。転送決定を行うための基準として、常により新しいタイムスタンプを有する経路が選ばれるが、必ずしも公示 (advertise) する必要はない。他にも代替経路が可能であるようなときは、最短距離を有する経路が記憶され、その他の経路は忘れられる。同じ距離の代替経路間で選択を行わなければならない場合は、最新のタイムスタンプを有する経路が選択される。ローカル経路指定テーブルを伝送する各移動コンピュータは、各伝送をその

ローカル時間値でスタンプするので、ほとんどすべてのタイムスタンプは最終宛先から発する。経路指定テーブルが伝播される自然な方法によって、他の各コンピュータにタイムスタンプが選ばれ、それらの他のコンピュータは、起点となる移動コンピュータに関する経路指定エントリを維持することを決定することができる。移動コンピュータが同期している場合は、移動コンピュータのアドホック共同体全体に対してタイムスタンプが1つだけ必要である。

【0022】以上述べたことは、事実上、必要な経路指定データを伝播し記憶する方法についてだけであつた。経路指定テーブルをローカルに操作し更新する実際の方法的詳細は後で述べる。データは、入力パケット用と出力パケット用の2つの方法で使用される。リンク層経路指定の動作は、リンク層と、もしある場合にはネットワーク層プロトコル(国際標準化機構(ISO)の成層ネットワーク用語における「層3」)との間に挿入される薄いプロトコル層で行われるものと考え、最もよく理解される。すなわちリンク層経路指定は、出力パケットについては、他のリンク層動作の前、任意の高水準プロトコル動作の後に行われる。反対に入力パケットについては、本明細書のリンク層経路指定動作は、他のリンク層動作(たとえば、フレーム指示、データ安全性の検査)の後、他の高水準プロトコル動作の前に実行される。

【0023】出力パケットについては、リンク層経路指定動作は実際の宛先が隣接するコンピュータか否かを判定し、そうでない場合は、リンク層が受け取ったデータはカプセル化され、新しい宛先アドレスと新しい層2

(リンク層)のパケット・タイプを含む新しいリンク層ヘッダが構築される。新しい宛先は、実際の宛先までの進路に沿った次ホップの宛先である。パケット・タイプは、ここに記載するリンク層経路指定手順を開始するための一般に合意された数である。すなわち、アドレス分解プロトコル(ARP)が要求しあるいは高水準プロトコル処理が開始されるのと同じ方式で、新しい経路指定要求の様々な処理が開始される。リンク層経路指定モジュールが受け取ったデータが新しいパケット・タイプと新しい宛先で包まれようがそうでなかろうが、パケットはリンク層の通常動作によって伝送される。

【0024】そのようなリンク層動作によって別の宛先に経路指定される必要があるパケットが入ってくると、そのパケットは次ホップに再アドレス指定されて送られる。次ホップが実際の宛先である場合は、経路指定のために必要な元のデータをカプセルから取り出しカプセルを廃棄することによって、実際の宛先と実際に必要なパケット・タイプが露出される。次ホップが実際の宛先ではない場合は、パケット・タイプとリンク層ヘッダは、見かけ上の宛先がその進路に沿った次ホップのアドレスに変更される以外はそのままになり、チェックサムまた

はデータ安全性標識が必要に応じて更新される。いずれの場合にも、リンク層経路指定を実行するコンピュータにおいて、リンク層より上のプロトコルは活動化されない。

【0025】図2は、両方向無線リンク50および移動ホストMH1ないしMH8を有するアドホック・ネットワーク10を示す。また、点線で示すようにMH2の隣の位置からMH7とMH8の隣の位置に移動するMH1が示されている。本発明によって、固定した有線ネットワークの基地局との通信なしに、ネットワーク10の移動ホスト間でパケットを経路指定することができる。ネットワーク10を介してパケットを経路指定するために必要な情報は、各移動ホストで維持される表(後に示す)に含まれる。これらの表は、移動ホストの移動によって引き起こされるネットワーク10の絶えず変化するトポロジを反映するように更新される。

【0026】この概念の総体は、各移動ホストにその経路指定テーブルを周期的に同報通信させ、それに対応して隣接局からそのような同報通信を受け取ったとき、その経路指定テーブルを更新することである。このようにして、各同報通信が処理されるとき、すべての移動ホストは、アドホック・ネットワークを確立することを望む協力するすべての移動ホスト間の相互接続の現トポロジの完全な記述を構築する。各経路指定テーブルのエントリはタイムスタンプでタグ付けされ、そのタイムスタンプはベルマン・フォード経路指定などの宛先ベクトル・アルゴリズムに付随するいくつかの問題を解決するために使われる。そのようなアルゴリズムは、計算上効率的である。経路は、所望の宛先に対して「最良の」距離をもつとき選ばれる。「距離」は通常、宛先に到達する前にパケットが飛び越さなくてはならない「ホップ」の数である。

【0027】同報通信において受け取った経路はまた、次にその経路指定情報を同報通信するときに受信側によって公示されるが、入力パケットが宛先に達するためにもう一回ホップ(すなわち、送信側から受信側までのホップ)を必要とするため、受信側は経路を公示する前に距離に増分を加える。

【0028】選択すべき最も重要なパラメータの1つは、経路指定情報パケットを同報通信する合間の時間である。しかし、移動ホストが新しい経路情報または実質上修正された経路情報を受け取ると、新しい情報がすぐに再伝送され、協力するすべての移動ホストの間で最も迅速な経路指定情報の流布を達成する。この即時の再同報通信は、プロトコルができるだけ早く収められなければならないという新しい要件を導入する。移動ホストの移動によって、同報通信のあらしが起こり、無線媒体の使用可能性が低下すると悲惨なことになる。

【0029】移動ホストは、所々に移動するとき破壊リンクを生じる。破壊リンクは「無限大」(すなわち、最

大許容距離よりも大きな任意の値)の距離で表される。次ホップまでのリンクが破壊されたとき、その次ホップを通るすべての経路に即座に無限大の距離が割り当てられ、更新されたタイムスタンプが割り当てられる。これは実質的な経路変化とみなされるため、同報通信の経路指定情報パケットにおいてそのような修正経路が即座に開示される。破壊リンクを記述するための情報の構築は、宛先移動ホスト以外の任意の移動ホストによってタイムスタンプが作成される唯一の状況である。また、移動コンピュータが同期されている場合は、1つのタイムスタンプだけが必要である。発生源である移動ホストによって定義されるタイムスタンプは偶数になるように定義され、無限大の距離を示すために作成されるタイムスタンプは奇数である。このようにして、任意の「実数」のタイムスタンプが、無限大の距離に取って代わる。

【0030】極めて大きい移動ホスト群においては、経路指定情報パケットの同報通信の合間の時間におそらく調整が行われる。これらのパケットで運ばれる情報の量を減らすために、2つのタイプを定義する。一方は、「全ダンプ」と呼ばれ、使用可能なすべての経路指定情報を運ぶ。他方のタイプは、「増分式」と呼ばれ、最後の全ダンプ以降に変化した情報だけを運ぶ。増分式経路指定更新は、設計により、1つのネットワーク・プロトコル・データ単位(NPDU)に収まるようになっている。比較的小さな移動ホスト群の場合でも、全ダンプにはおそらく複数のNPDUが必要となる。移動ホストの移動が起こらないときは、比較的まれに、全ダンプを送ることができる。移動が頻繁になり、増分の大きさがNPDUの大きさに近づくと、(次の増分が少なくなるように)全ダンプをスケジューリングすることができる。

【0031】移動ホストが新しい経路指定情報を(通常は、前述の増分式パケットで)受け取ると、その情報を、前の経路指定情報パケットからあらかじめ入手可能な情報と比較する。より新しいタイムスタンプを有する経路が使用される。古いタイムスタンプを有する経路は破棄される。既存の経路と同じタイムスタンプを有する経路が「より良い」距離を有する場合は、その経路が選ばれ、既存経路は破棄され、または余り好ましくないものとして記憶される。新たに受け取った同報通信情報から選ばれた経路の距離は、それぞれ1ホップだけ増分される。新たに記録された経路、または改善された距離を示す経路は、現移動ホストの隣接局に即座に公示するようにスケジューリングされる。

【0032】様々な移動ホスト間のタイミング・スキューが予期される。移動ホストによる経路指定情報の同報通信の合間の期間は、ある程度の規則正しさが予期される場合でも、多少非同期な事象とみなされる。そのように独立して伝送するエージェント群では、経路を更新するために上記手順を使うことによって、若干の変動が発

生することがある。宛先移動ホストが移動しなかったときでも、ある次ホップから別の次ホップまでの経路を一貫して変化させるパターンで、特定の移動ホストが新しい経路指定情報を受け取るという問題が生じることがある。これは、選ぶべき新しい経路に関して、2つの進路すなわち遅いタイムスタンプを有するものより良い距離を有するものがあるからである。移動ホストは常に、より新しいタイムスタンプとを有する、同じ宛先までの2つの経路を(異なる隣接局を介して)次から次へと受け取ると想像されるが、常に最初は悪い距離の経路を得る。注意しないと、これは、その宛先からの新しいタイムスタンプごとに、新しい経路伝送の連続バーストを引き起こす。新しい距離はそれぞれ、近くにあるすべての移動ホストに伝播され、さらにその隣接局がその隣接局に伝播し、以下同様に続く。

【0033】本発明の好ましい実施例による解決法は、より良い距離を有する経路がまもなく現れそうだと移動ホストが判定できるときに、そのような経路の公示を遅延することである。遅いタイムスタンプを有する経路が使用可能でなければならないが、前に到達できなかった宛先への経路でない限り、即座に公示する必要はない。したがって、各移動ホストで保持される経路指定テーブルは2つあり、一方は転送パケットと共に使用するためのものであり、他方は増分式(および全)経路指定情報パケットを介して公示されるものである。より良い距離を示す経路指定情報の到着が近い確率を判定するために、移動ホストは、より良い距離によって更新される前に、一般に特定の経路が継続する長さの履歴を保持しなければならない。

【0034】上記手順はすべて、プロトコル・スタックのネットワーク層(層3)とリンク層(層2)のどちらで実行されようとも有効である。したがって、層3のいくつかの可能なプロトコルを使って通信できる移動ホストのアドホック・ネットワークを設けたい場合、パケット同報通信と転送を層2で実施することができる。これにより、たとえば、中間移動ホストと同じ層3のプロトコルを実施しなかった2つの端点のサービスにおいて、中間移動ホストがパケットを転送できるようになる。

【0035】経路指定テーブルに記憶されたアドレスは、このアドホック・ネットワーク・プロトコルが実施される層に対応する。つまり、層3における動作は次ホップ用のネットワーク層アドレスおよび宛先アドレスを使用し、層2における動作は層2の媒体アクセス制御(MAC)アドレスを使用する。

【0036】しかし、転送テーブルのためにMACアドレスを使用すると新しい要件が導入される。困ったことに、層3ネットワーク・プロトコルはネットワークの3つのアドレスに基づいて通信を実現するが、これらの層3アドレスをMACアドレスに分解する方法を提供しなければならない。そうでなければ、多数の異なるアドレ

ス分解機構が設けられ、その分解機構を利用するときは常に、無線媒体において対応する帯域幅の損失が認められることになる。このことは重要であり、その理由は、そのような機構はアドホック・ネットワークのすべての移動ホストによる同報通信と再伝送同報通信を必要とすることになるからである。したがって特別な注意を払わないと、アドレス解決はすべて、ネットワークの標準動作における不調のように見え、それはどの活動ユーザにもはっきりとわかるであろう。

【0037】本発明による解決法は、層2における動作に関して、層2の経路指定情報に加えて層3プロトコル情報を含めることである。各宛先ホストは、層3のどのプロトコルを支援するかを公示し、宛先への到達可能性

宛先アドレス
次ホップ・アドレス
距離
タイムスタンプ
導入時間
安定度データへのポインタ
プロトコル・データへのポインタ

【0040】たとえば、図1における移動ホスト4を検討する。各移動ホストのアドレスをMHXで表し、すべての移動ホストがインターネット・プロトコル(IP)だけで支援されると仮定する。さらにすべてのタイムスタンプがTNNN_MHXで示されると仮定する。このMHXはタイムスタンプを作成したコンピュータを指定し、TNNNは時間の値である。また、移動ホスト1が移動ホスト2から移動する前に、他のすべての移動ホス

| 宛先 | 次ホップ | 距離 | タイムスタンプ | 導入 | フラグ | 安定データ | プロトコル・データ |
|-----|------|----|----------|----------|-----|----------|-----------|
| MH1 | MH2 | 2 | T406_MH1 | T001_MH4 | | PTR1_MH1 | PTR2_MH1 |
| MH2 | MH2 | 1 | T128_MH2 | T001_MH4 | | PTR1_MH2 | PTR2_MH2 |
| MH3 | MH2 | 2 | T564_MH3 | T001_MH4 | | PTR1_MH3 | PTR2_MH3 |
| MH4 | MH4 | 0 | T710_MH4 | T001_MH4 | | PTR1_MH4 | PTR2_MH4 |
| MH5 | MH6 | 2 | T392_MH5 | T002_MH4 | | PTR1_MH5 | PTR2_MH5 |
| MH6 | MH6 | 1 | T076_MH6 | T001_MH4 | | PTR1_MH6 | PTR2_MH6 |
| MH7 | MH6 | 2 | T128_MH7 | T002_MH4 | | PTR1_MH7 | PTR2_MH7 |
| MH8 | MH6 | 3 | T050_MH8 | T002_MH4 | | PTR1_MH8 | PTR2_MH8 |

【0042】これから、たとえば、ほとんどのコンピュータの導入時間がほぼ同じなので、すべてのコンピュータがほぼ同じ時間にMH4にとって使用可能になったと推定される。また、すべてのタイムスタンプ・フィールドが単位桁が偶数の時間を有するので、コンピュータ間のリンクは1つも破壊されなかったと推定される。図1には、他の経路で置き換えられそうなまたは競合しそうな特定宛先への経路がないので、PTR1_MHXはすべてヌル構造へのポインタとなる。プロトコル・データ・ポインタはすべて、以下のフォーマットを有する構造

宛先アドレス
距離
タイムスタンプ

を公示する各移動ホストは、その公示の他に、宛先で支援される層3プロトコルに関する情報を含むことになる。この情報は、変化したときにだけ伝送すればよく、それはめったに起こらない。この情報は、各「全ダンプ」の一部として伝送されることになる。各移動ホストは、いくつか(あるいは多数の)の層3プロトコルを支援し得るので、このリストは長さが可変でなければならないことになる。

【0038】表1に、図2に示したネットワーク10内の各移動ホストで維持される内部転送テーブルにおける経路エントリの構造を示す。

【0039】

【表1】

プロトコル依存サイズ
プロトコル依存サイズ
符号なしの整数
宛先から;符号なしの整数
装置依存(たとえば、32ビット)
装置依存
装置依存、層2のみ

トにタイムスタンプTNNN_MHXを有するエントリがあると仮定する。このとき、MH4における内部転送テーブルは次のようになる(行は様々な移動ホストに対応し、列は前記構造で記述されたデータに対応することに留意されたい)。

【0041】

【表2】

| 導入 | フラグ | 安定データ | プロトコル・データ |
|----------|-----|----------|-----------|
| T001_MH4 | | PTR1_MH1 | PTR2_MH1 |
| T001_MH4 | | PTR1_MH2 | PTR2_MH2 |
| T001_MH4 | | PTR1_MH3 | PTR2_MH3 |
| T001_MH4 | | PTR1_MH4 | PTR2_MH4 |
| T002_MH4 | | PTR1_MH5 | PTR2_MH5 |
| T001_MH4 | | PTR1_MH6 | PTR2_MH6 |
| T002_MH4 | | PTR1_MH7 | PTR2_MH7 |
| T002_MH4 | | PTR1_MH8 | PTR2_MH8 |

を指す。プロトコルID=IP、プロトコル・アドレス長=4バイト、プロトコル・アドレス=[MHX, Net.

addr.ess]
ここで、MH1.Net.addr.essは、標準のInternet4オクテット形式で表示されたMH1に関する4バイトのIPアドレスである。

【0043】表3は、公示された経路テーブルにおける経路エントリの構造を示す。

【0044】

【表3】

プロトコル依存サイズ
符号なしの整数
宛先から;符号なしの整数

次の層3アドレスのサイズ 8ビット、0のときはもうない
 次アドレスのプロトコルID 8ビット
 次の層3プロトコルのアドレス

【0045】最後の項目は、層2でアドホック・アルゴリズムが動作するときにだけ現れる。次ホップはすべての公示に暗示されるので、リストする必要はない。動作が層2で行われ、移動ホストはアドレスX:X:X:X:X:X:X:Xを有し、したがって移動ホストはMACアドレス1:1:1:1:1:1:1:1 (標準フォーマットで示した) を有すると仮定する。さらにIPは、層3のプロト

コルID7を有するものとして示されると仮定する。またそれと対応して、移動ホストMHXのインターネット・アドレスは、X.X.X.X.と表されると仮定する。そこで、上記の状況において、公示される経路は以下のように表される。

【0046】

【表4】

| 宛先 | 距離 | タイムスタンプ | 長さ | ID | 層3アドレス | 長さ |
|-------------|----|----------|----|----|---------|----|
| 1:1:1:1:1:1 | 2 | T406_MH1 | 4 | 7 | 1.1.1.1 | 0 |
| 2:2:2:2:2:2 | 1 | T128_MH2 | 4 | 7 | 2.2.2.2 | 0 |
| 3:3:3:3:3:3 | 2 | T564_MH3 | 4 | 7 | 3.3.3.3 | 0 |
| 4:4:4:4:4:4 | 0 | T710_MH4 | 4 | 7 | 4.4.4.4 | 0 |
| 5:5:5:5:5:5 | 2 | T392_MH5 | 4 | 7 | 5.5.5.5 | 0 |
| 6:6:6:6:6:6 | 1 | T076_MH6 | 4 | 7 | 6.6.6.6 | 0 |
| 7:7:7:7:7:7 | 2 | T128_MH7 | 4 | 7 | 7.7.7.7 | 0 |
| 8:8:8:8:8:8 | 3 | T050_MH8 | 4 | 7 | 8.8.8.8 | 0 |

【0047】ここで、移動ホスト1が移動ホスト5および7の周辺に移動し、他の移動ホスト（特に移動ホスト2）から離れたと仮定する。移動ホスト4における新しい内部転送テーブルは以下になる。

【0048】

【表5】

| 宛先 | 次ホップ | 距離 | タイムスタンプ | 導入 | フラグ | 安定データ | プロトコルデータ |
|-----|------|----|----------|----------|-----|----------|----------|
| MH1 | MH6 | 3 | T516_MH1 | T810_MH4 | M | PTR1_MH1 | PTR2_MH1 |
| MH2 | MH2 | 1 | T238_MH2 | T001_MH4 | | PTR1_MH2 | PTR2_MH2 |
| MH3 | MH2 | 2 | T674_MH3 | T001_MH4 | | PTR1_MH3 | PTR2_MH3 |
| MH4 | MH4 | 0 | T820_MH4 | T001_MH4 | | PTR1_MH4 | PTR2_MH4 |
| MH5 | MH6 | 2 | T502_MH5 | T002_MH4 | | PTR1_MH5 | PTR2_MH5 |
| MH6 | MH6 | 1 | T186_MH6 | T001_MH4 | | PTR1_MH6 | PTR2_MH6 |
| MH7 | MH6 | 2 | T238_MH7 | T002_MH4 | | PTR1_MH7 | PTR2_MH7 |
| MH8 | MH6 | 3 | T160_MH8 | T002_MH4 | | PTR1_MH8 | PTR2_MH8 |

【0049】MH1のエントリだけが新しい距離を示しているが、間の時間には、多くの新しいタイムスタンプ・エントリを受け取っている。すなわち、最初のエントリは、フラグM（距離MetricのM）を有し、次の全ダンプが生じるまでに、続いて起こる増分経路指定情報の更新において公示されなければならない。移動ホスト1が移動ホスト5と7の周辺に移動したとき、増分式経路指定情報の即時更新を開始した。これはその後、移動ホスト6に同報通信された。また、移動ホスト6は、重要で新しい経路指定情報を受け取ったと判断し、移動ホスト1に関する新しい経路指定情報を移動ホスト4に運ぶ即時更新を開始した。移動ホスト4は、この情報を受け取ると、次の経路指定情報の全ダンプまで、それをすべての間隔で同報通信することになる。移動ホスト4において、増分式公示経路指定の更新は以下の形式を有する。

【0050】

【表6】

宛先 距離 タイムスタンプ

| | | |
|-------------|---|----------|
| 4 4 4 4 4 4 | 0 | T820_MH4 |
| 1 1 1 1 1 1 | 3 | T516_MH1 |
| 2 2 2 2 2 2 | 1 | T238_MH2 |
| 3 3 3 3 3 3 | 2 | T674_MH3 |
| 5 5 5 5 5 5 | 2 | T502_MH5 |
| 6 6 6 6 6 6 | 1 | T186_MH6 |
| 7 7 7 7 7 7 | 2 | T238_MH7 |
| 8 8 8 8 8 8 | 3 | T160_MH8 |

【0051】この公示では、移動ホスト4が公示を行っているので、移動ホスト4に関する情報が最初になる。移動ホスト1は、低いアドレスを有するためではなく、影響を及ぼす重要な経路変更を有するただ1つのものなので、移動ホスト1に関する情報が次になる。増分式経路指定の更新全体は、以下のような形式を有する。

【0052】

【表7】

伝送情報 「自局アドレス」、距離≡0
 層3のプロトコル使用可能度情報が変更された経路
 距離が変更された経路

タイムスタンプが変更された経路

【0053】この例では、層3プロトコル構成を変更した移動ホストはない。1つのコンピュータが新しい位置にあるため、その経路指定情報を変更した。すべてのコンピュータは、新しいタイムスタンプを最新に伝送した。更新されたタイムスタンプが多すぎて単一のバケットに収まらない場合は、収まるタイムスタンプだけが伝送される。これらは、いくつかの増分更新間隔にわたって公平に伝送するために選択される。

【0054】全経路指定情報バケットの伝送には、そのようなフォーマットは必要でない。必要な数のバケットが使用され、(必要な層3アドレス情報を含めて)使用可能な情報がすべて伝送される。

【0055】アドホック・ネットワーク・プロトコル内で時間依存性のいくつかの動作を処理するために、標準の事象リスト構造を維持しなければならない。ノードの例は、以下のようでもよい。

【0056】

【表8】

事象時間

事象識別

事象データ (経路エントリへのポインタ)

【0057】コンピュータのクロックが刻時するとき、事象リストが検査される。最初のノードが満了した場合、リストと、修正処理手順を呼ぶために使用される識別と、事象処理ルーチンへの引数として引き渡される事象データとから事象ノードが引き出される。

【0058】以下の説明は、安定待ち待ち時間テーブルに関し、経路指定テーブル・エントリの変動を防ぐ際のその使用法を説明する。経路更新は以下の基準に従って選択されるために、全般的な問題が生じる。

・タイムスタンプがより新しい場合、その経路は常に好ましい。

・そうでなければ、タイムスタンプが同じではあるが距離がより良い (短い) 場合に、その経路は好ましい。

問題を理解するために、識別タイムスタンプを有する2つの経路を、間違った順序で移動ホストが受け取ったと仮定する。すなわち、移動ホスト4が、最初により長い距離の次ホップを受け取り、そのすぐ後にタイムスタンプが同じで短い距離の別の次ホップを得ると仮定する。

これは、それほど規則的でなく更新を伝送する多数の移動ホストがあるときに起こり得る。その代わりに、移動ホストが著しく異なる伝送間隔でまったく無関係に動作している場合は、それに対応してより少ないホストでこの状況が起こり得る。いずれにせよ、図3において、共通の宛先MH9に両方とも接続されているが他の移動ホストは共通でない、2つの別々の移動ホスト群中に、この問題を引き起こすのに十分な移動ホストがあると仮定する。さらに、すべての移動ホストは約15秒毎に更新を伝送しており、移動ホストMH2はMH9まで12ホ

ップの経路を有し、移動ホストMH6はMH9まで11ホップの経路を有すると仮定する。さらに、MH2からの経路指定情報の更新は、MH6からの経路指定情報の更新よりも約10秒前にMH4に到着すると仮定する。これは、移動ホストMH9から新しいタイムスタンプが発行されるたびに行われる。実際には、たとえば、新しいタイムスタンプ更新を有するホストが多すぎて全ホストが単一の増分バケットの更新内に収まらないときに起こるように、群IIの任意の移動ホストが複数の増分式更新間隔でそのタイムスタンプ更新を発行し始める場合に、時間差が激烈になり得る。一般に、ホップ数が大きくなるほど、図3における更新配送の差が激烈になると予期される。

【0059】安定待ち時間データは、最初の2つのフィールドによって指定される、以下の形式を有するテーブルに記憶される。

【0060】

【表9】

宛先アドレス

20 次ホップ・アドレス

最終安定待ち時間

平均安定待ち時間

【0061】新しい経路指定情報の更新が移動ホスト4に達すると仮定する。新しいエントリのタイムスタンプは、今使用したエントリ内のタイムスタンプと同じであり、より新しいエントリはより悪い (すなわち、より長い) 距離を有する。そこで、移動ホスト4は、次の転送決定を行う際に新しいエントリを使用しなければならない。ただし、移動ホスト4は、新しい経路を即座に公示する必要はなく、公示する前にどれだけ待つかを決めるためにその経路の安定待ち時間テーブルを調べることができる。この決定のために平均安定待ち時間を使用する。たとえば、経路を公示する前に、移動ホスト4は (平均安定待ち時間 * 2) 遅延することを決定できる。

【0062】これは非常に有益なことである。というのは不安定な可能性がある経路が即座に公示されると、その結果がネットワークを介して波及し、移動ホストMH9のタイムスタンプの更新がアドホック・ネットワークを介して波及するたびに、この悪い結果がおそらく繰り返されることになるからである。一方、移動ホストMH6を経由するリンクが本当に壊れた場合は、MH2を経由する経路の公示をすぐに行うべきである。これを達成するために、移動ホストMH4に変動の履歴があるときは、群II内の中間ホストが問題を発見し、移動ホストMH9までの進路に沿った経路に関して無限の距離を示すトリガされた増分更新を開始するように、十分早くリンク破壊を検出すべきである。すなわち、以前に起こった経路指定更新の変動に類似する問題が現れる場合、その問題は、変動を回避する機構を無力にするのに十分な時間経路更新パターンを支配する他の効果を有するように

思われる。さらに、無限大の距離を有する経路は、定義によりすぐに公示しなければならない。

【0063】ダンピング機構を最新の事象に有利に片寄らせるために、特定経路の安定待ち時間の最新の測定値を、古い測定値よりも大きな加重因子でカウントしなければならない。そして、重要なことに、経路が真に安定状態とみなされる前にどれだけ安定状態でなければならないかを示すパラメータを選択しなければならない。これは結局、安定待ち時間テーブルにおける1対のアドレス(宛先、次ホップ)の、安定待ち時間の最大値を指定することになる。この最大値を有する経路よりも安定な経路は、それが異なる次ホップまたは距離を有する別の経路で置き換えられる場合と、トリガされた更新を引き起こす。

【0064】リンク層ソフトウェアが経路指定テーブル管理を実施する方法は周知であるが、説明のために、特定の実施態様について若干の詳細を示す。このテーブル自体は、オペレーティング・システムのデータ・メモリ内に記憶されたデータでしばしばそうであるように、静的に割り振られた固定サイズ・エントリのアレイである。各エントリは「次の」エントリを指定する整数フィールドを有し、これにより通常モードの経路指定テーブルへのアクセスが、線形の探索によるのではなく(静的に大きさが決められたアレイの場合によくあることだが)、リンクしたリストのアクセスのようになる。各宛先ノードは、せいぜい3つ代替経路しか有することができない。これらの経路は、最適経路を最初にして、リストの3つの連続する要素として記憶される。示された最適経路が失敗したり、そのデータが不調と判断された場合は、次の経路が実際に「昇進(promoted)」する。

【0065】更新がテーブルに適用されると同時に「隣接局」から新しい経路指定更新を受け取ったとき、不調のエントリを削除するための処理も行われる。不調のエントリは、最近の二三の更新期間内に更新が適用されなかったエントリと定義される。各隣接局は定期的に更新を送ると期待されるので、しばらく更新を受け取らないと、受信側は対応するコンピュータがもはや隣接局でないと判断することがある。それが起こると、そのコンピュータを実際の(かつては隣接の)宛先として示す経路を含めて、そのコンピュータを次ホップとして使用するどの経路も削除される。エントリが決定される前に発生する更新期間の数が多いと、不調の経路指定エントリが増すことになるが、伝送エラーも増す。伝送エラーは、多くの無線実施態様の場合によくあり得ることだが、CSMA型の同報通信媒体を使用するときにも発生する可能性が高い。リンクが壊れたときは、そのリンクおよびそのリンクに依存する経路に対して、無限大の距離の経路をスケジュールすべきである。

【0066】図4は、データが不調であると判定された

ときのタイム・アウト手順を示す。まず機能ブロック40で、事象リスト・データから経路エントリが得られ、ついで機能ブロック41で、内部テーブルから経路が削除される。機能ブロック42で、公示される経路のテーブルに無限大の距離が挿入され、次に判断ブロック43で、その宛先が他の宛先の次ホップかどうかを判定する試験が行われる。そうである場合は、機能ブロック44で、現在到達不可能な宛先に対して無限大の距離の経路が公示される。図5に示したように、"ADVERTISE"事象処理は、機能ブロック45で公示された経路リストに指定された経路を挿入する処理と、次いで機能ブロック46でINCREMENTALフラグをセットする処理とを含む。同時に、SHOWN YETフラグがリセットされる。挿入される経路は、宛先まで無限大の距離を示すことになる。

【0067】図6は、移動ホストからの増分式更新伝送の論理を示す。この処理は、機能ブロック47から始まり、プロトコル使用可能度の変化が挿入される。次いで、機能ブロック48で公示経路リストが走査され、判断ブロック49でFLAGSおよびSHOWN YETが0であるか検査が行われる。この条件に合えば、機能ブロック50で、その経路が挿入され、フラグがセットされる。次に、判断ブロック51で試験が行われ、出力パケットが一杯過ぎないかどうか判定する。一杯過ぎる場合は、処理が終わる前に、機能ブロック52で全ダンプがスケジュールリングされ、そうでない場合は、機能ブロック48に戻って公示経路リストを走査する。

【0068】公示経路リストが走査されると、判断ブロック53で試験が行われ、FLAGSとINCREMENTALが0でないかどうか判定する。0である場合は、処理は機能ブロック48に戻り、そうでない場合は、機能ブロック54でその経路が挿入される。判断ブロック55で試験が行われ、出力パケットが一杯過ぎないかどうか判定する。一杯過ぎる場合は、機能ブロック52で全ダンプがスケジュールリングされ、そうでない場合は、機能ブロック56で公示経路リストが再び走査されるが、このときはLAST ADV TIMESTAMPから始まる。公示経路リストが走査されるとき、判断ブロック57で試験が行われ、FLAGSおよびNEW TIMESTAMPが0でないかどうか判定する。0でない場合は、機能ブロック58で経路が挿入され、判断ブロック59で試験が行われ、出力パケットが一杯過ぎないかどうかを判定する。一杯過ぎる場合は、機能ブロック60でLAST ADV TIMESTAMPが提示された最後の経路にセットされ、そうでない場合は、機能ブロック61でLAST ADV TIMESTAMPがゼロにセットされ、処理が終了する。

【0069】図7は、移動ホストからの全ダンプ伝送の論理を示す。まず、判断ブロック62で試験が行われ、またどの経路も示されていないかどうかを判定する。示

されていない場合は、処理が終わる前に機能ブロック63で、増分伝送が行われ、全ダンプが再びスケジュールリングされる。そうでない場合は、機能ブロック64で、指定されたテーブル・フォーマットに従って使用可能なプロトコルがすべて挿入される。次に、機能ブロック65で、テーブルの形式に従って、公示経路がすべて挿入される"FLAGS"フィールドが削除される。最後に、機能ブロック66で、すべての公示経路で増分がリセットされ、処理が終了する。

【0070】図8は、受信時の全ダンプ処理の論理を示す。まず機能ブロック67で入力データが走査され、判断ブロック68でタイムスタンプが新しいかどうか、判断ブロック69で経路が新しいかあるいは新しい距離を有するかどうか、また判断ブロック70でどれかのプロトコルが変更されたかどうかを判定する。タイムスタンプが新しい場合は、機能ブロック71で、内部経路指定テーブルに現行値が置かれ、タイムアウト事象が再びスケジュールリングされて、テーブル内で新しいタイムアウトがマークされる。次に、機能ブロック72で、その経路の安定待ち時間の測定を開始し、処理が終了する。一方、経路が新しいかあるいは新しい距離を有する場合は、機能ブロック73で、エラー・アクティビティがスケジュールリングされる。次いで、機能ブロック74で、安定待ち時間が更新され、処理が終了する。一方、どれかのプロトコルが変化した場合は、機能ブロック75でたとえばARPテーブル管理を使って、適切な層3アクティビティが変更される。

【0071】図9は、受信時の増分式更新処理を示す。機能ブロック76で入力データが走査され、判断ブロック77でプロトコル使用可能度が変更されたかどうか、判断ブロック78で経路が新しいかどうか、判断ブロック79でタイムスタンプが古いかどうか、判断ブロック80でタイムスタンプが同じで距離がより短いかどうかを判定する。プロトコル使用可能度が変化している場合は、機能ブロック81で適切な層3処理ルーチン呼び出し、判断ブロック78に進む。経路が新しい場合は、機能ブロック82で出力増分更新がスケジュールリングされ、処理が終了する。タイムスタンプが古い場合は、判断ブロック83でさらに試験が行われ、経路が無限大の距離を有するかどうか判定する。経路が無限大の距離を持たない場合は、機能ブロック84で破棄され、処理が終了する。経路が無限大の距離を有する場合は、機能ブロック85でSHOWN YETフラグがリセットされ、処理が終了する。タイムスタンプが同じで距離がより短い場合、機能ブロック86で現安定待ち時間が更新され、機能ブロック87で公示リストに新しいエントリが最初に入れられる。次に、機能ブロック88で、SHOWN YETフラグがリセットされ、増分がセットさ

*/

struct advertised route entry{

れ、関係する"ADVERTISE"事象が削除され、処理が終了する。判断ブロック80に戻り、結果が否定の場合は機能ブロック89に進み、内部テーブル内の経路エントリが使用され、タイムアウトがリセットされる。次に、機能ブロック90で、安定待ち時間の現行評価後、覚醒がスケジュールリングされ、処理が終了する。

【0072】図10は、安定待ち時間が過ぎた後に、経路を公示経路に挿入するための流れ図を示す。この処理は、図9の機能ブロック90でセットされた覚醒タイマが、機能ブロック101で鳴ると始まる。これが起こると、判断ブロック102で試験が行われて、確立された経路が公示された経路と同じであるかどうかを判定する。同じ場合は、何も行う必要はなく、処理は終了する。しかし異なる場合は、機能ブロック103でSHOWN YETがリセットされ、処理が終わる前に、機能ブロック104で次の増分更新がスケジュールリングされる。

【0073】上記以外にも、関与する各コンピュータ（移動局または基地局）によって同報通信される経路指定テーブル内の各エントリの一部分として伝送される追加のデータ・フィールドがある。これらのフィールドは、たとえば、高水準のプロトコルまたはリンク層の動作に依存する他のプロトコルによって決まる。たとえば、正しいARP動作を可能にするために、各経路指定テーブル・エントリが、宛先アドレスに対応するインターネット・プロトコル(IP)アドレスも含まなければならないこともある。これは、隣接局のために経路指定機能を利用するとき、中間のコンピュータを使用可能にするために行われ、ARP同報通信の経路指定の代わりに「代理ARP」も発行する。

【0074】協力する移動ホスト間でアドホック・ネットワークを達成するために使用される、様々な手順を記述する疑似Cコードを以下にリストする。

【0075】

```
struct forwarding route entry{
    address t destination;

    address t next hop;
    value t metric;
    value t settling time;
    value t install time;
    protocol list;
    flags;
}
```

このテーブルは、これらの手順を実行する移動ホストに関するデータを常に含むように初期設定する。

23

24

```

address t destination;
value t metric;
proto ptr protocol list;
struct advertised route entry
    *advertised route table={myaddress, 0, my protocol list};
struct protocol list{
    value t protocol type;
    value t address size;
    u char[] layer 3 address;
}
/*
*各移動ホストは、経路エントリの2つのテーブルを維持しなければならない。
*一公示される経路エントリ
*一転送用に使われる経路
*/
/*
*各移動ホストは、可能な各種のタイムアウト事象に関してノードを有する事象
リ
*ストを維持しなければならない。可能な事象は以下のとおりである。
*一経路指定テーブル・エントリをタイムアウトする
*一起こり得る変動を回避するためにその公示が遅延された公示テーブルへの経
路
* を加える。
*一公示を周期的に同報通信する（増分式または全）。
*/
Timeout()
{
    get event from list();
    switch (event type)
    case ROUTE TIMEOUT:
        bad route=event type->route;
        if (bad route->metric =1) /* おっと、隣接局が死んだ*/
            for (route = first; /*テーブル内のすべての経路について*/){
                if route->next hop = bad route){
                    route->metric = INFINITE METRIC;
                    route->flags = METRIC CHANGED;
                    route->timestamp |=1;
                }
            }
        bad route->flags |=METRIC CHANGED
        bad route->metric=INFINITE METRIC;
        bad route->timestamp|=1; /* 1だけ増分した*/
        do increment();
        break;
    case ADD ADVERTISEMENT:
        route->flags |=CHANGED;
        break;
    case DO ADVERTISEMENT:
        if (full dump scheduled)
            do full dump();

```

```

25
else
    do incremental();
    break;
}
}
struct settling time table{
    address t destination;
    time t settling time;
    value t number of next hops;
    addr list next hop list;
}
struct next hop list{
    address t next hop;
    list ptr *next hop list;
}
do full dump()
{
    get empty NPDU();
    for (/*公示された経路指定テーブル内の各経路*/)
    {
        copy route into NPDU()
        if (NPDU full) {
            transmit NPDU();
            get empty NPDU();
        }
        route->flags &= /*変更ビットをリセットする*/
            NOT(CHANGED | CHANGED PROTOCOL |
                CHANGED METRIC);
    }
    /*上記で自局データが自動的に含まれたことに留意されたい*/
    schedule full dump(USUAL PERIOD);
}
/*
*増分ダンプは、次のようないくつかの部分に集積する。
*一送信局（すなわち、この移動ホスト）に関するエントリ。
*一新しい移動ホストまたは層3プロトコル使用可能度情報を修正した移動ホス
ト
* に関するエントリ。
*一経路指定情報が実質上変更された新しい移動ホストに関するエントリ。
*一新しいタイムスタンプだけを反映しているエントリ。
*増分パケットは、1つの層3パケット（ネットワーク・プロトコル・データ・
ユ
*ニット）だけに収まるように制約される。重要な変化が多すぎて報告できない
場
*合は、全ダンプをスケジューリングしなければならない。タイムスタンプ更新
が
*多すぎて報告できない場合は、最後に報告されたタイムスタンプ更新を追跡し
、
*次回はそれからスタートする。
*/

```



```

do incremental dump()
{
    get empty NPDU();
    /*最初に自局の新しいタイムスタンプを送送する。距離0*/
    if (route to myself->timestamp & 0x00000001 !=0)
        printf ("予期しない内部タイムスタンプ error\r\n");
    route to myself->timestamp = route to myself +2;
    copy route into NPDU(route to myself)
    for ( route=first; /*公示経路指定テーブル内の各経路*/
        if (route->flags & PROTOCOL CHANGED)
            copy route into NPDU(route)
            if (NPDU full) (
                schedule full dump(IMMEDIATE);
                printf (Unexpectedly full incremental!\n");
                transmit NPDU();
                get empty NPDU();
            )
        }
    for ( route=first; /*公示経路指定テーブル内の各経路*/ {
        if (route->flags & ALREADY DONE) continue;
        /*上記でこれはすでに示された*/
        if (route->flags & METRIC CHANGED)
            copy route into NPDU(route)
            if (NPDU full) (
                schedule full dump(IMMEDIATE);
                printf ("予期せず全増分incremental!\n");
                transmit NPDU();
                get empty NPDU();
            )
        }
    }
    for (route=last timestamp shown; /* */) {
        if (route->flags & ALREADY DONE) continue;
        /*上記でこれはすでに示された*/
        if (route->flags & TIMESTAMP CHANGED) {
            copy route into NPDU(route)
            last timestamp shown = route;
        }
        if (NPDU full) (
            break; /*増分パケットの構築を停止する*/
        )
    }
    transmit NPDU();
}
/*
*移動ホストが隣接局のうちの1つから経路指定情報を受け取ると、パケットの
各
*経路エントリを調べて、その内部経路指定テーブルを更新するかどうかを決定
す
*る。入力パケットが全ダンプである場合は、通常、実質上異なる情報は期待さ
れ

```

*ない。より新しい場合あるいは新しさは同等だがより短い距離を有する場合に

*新しい経路が受け入れられる。

*

*入力経路がより新しいタイムスタンプという基準に基づいて受け入れられる場合

*、新しい経路を公示するかどうか決定を行う。この決定は、特定の宛先移動ホス

*トに関してパケットの送信側（すなわち、次ホップ）から得た経路の過去の履歴

*に依存する。

*

*次ホップを通る経路が次ホップまでの経路よりも（この移動ホストから次ホップ

*までのホップなので）1ホップだけ長くなることを反映するために、任意の入力

*経路に関して示された距離を1だけ増分しなければならないことに留意されたい

*。

*/

process incoming route update()

{

 must schedule incremental = FALSE;

 for (new route=packet data; /*パケット内の各エントリを行う*/(

 old route = find(new route, route table);

 new route->metric =new route->metric +1;

 if (new route->timestamp >

 old route->timestamp) {

 new route->timeout =

 calculate timeout(new route);

 replace (old route) new route);

 route table);

 delete timeout event (old route);

 schedule timeout event (new route);

 if (new route->type & CHANGED PROTOCOL){

 must schedule incremental = TRUE;

 new route->flags |=CHANGED PROTOCOL;

 new route->install time =

 current time();

 }

 else if ((new route->type &

 CHANGED METRIC) ||

 (new route->metric < old route->metric {

 must schedule incremental = TRUE;

 new route->flags |=CHANGED METRIC;

 new route->install time =current time();

 }

 else if (new route->metric >=

 old route->metric) {

 stable time = check settling time

31

32

```

        (new route)
        if (0 == stable time) /*それを公示する*/
            new route->flags |= CHANGED;
        else
            enter event list(new route,
                stable time, ADD ADVERTISEMENT);
    }
}
else if ((new route->timestamp ==
    old route->timestamp) &&
    (new route->metric < old route->metric)) {
    enter settling time data (old route,
        new route);
    new route->timeout =
        calculate timeout(new route);
    replace (old route, new route,
        route table);
    delete timeout event (old route);
    schedule timeout event (new route);
    new route->flags |= CHANGED METRIC;
    new route->install time = current time();
}
}
if (must schedule incremental) {
    schedule incremental(IMMEDIATE);
if (incoming packet->packet type == FULL DUMP)
    printf ("Full dump has new, unreported
        data!\n");
}
}
/*
*変動を弱めるために、経路変化の頻度に関するデータを保持する。ある状況で
は
*、同じタイムスタンプを有する特定の移動ホストで経路公示を受け取るが、「
順
*序外れ」で受け取る、すなわち同じタイムスタンプを有する2つの経路がより
短
*い距離に最初に達することが考えられる。
* 最終安定待ち時間
* 平均安定待ち時間
*2つのデータが保持される。
*平均安定待ち時間は、前の平均を16倍し、現行値を2倍し、それらの結果を
加
*えて18で割って計算される。これは、前の16個の結果よりも最新の結果に
少
*し大きな重みを与える効果がある。
*/
enter settling time (new route, old route)
{
    route data = find (new route->destination,

```

```

33      old route->next hop, settling time table);
      settling time = (16*
      route data->settling time) +
      2* (current time() - old route->install time);
      settling time = settling time ¥ 18;
    }
    check settling time (new route)
    {
      route data = find (new route,
      settling time table);
      settling time data =route data->settling time;
      if (settling time data == NULL) return 0;
      if (settling time data > MAXIMUM DELAY) return 0;
      else return (settling time data);
    }

```

【0076】本発明を1つの好ましい実施例に基づいて説明したが、添付の特許請求の範囲の趣旨および範囲内で本発明を修正して実施できることが、当業者には理解されよう。本発明の新しい経路指定アルゴリズムは、特に、移動コンピュータの動作に最も劇的に必要とされる「アドホック・ネットワーク」の作成を可能にするため20に開発されたものである。しかし、この経路指定アルゴリズム自体、および「アドホック・ネットワーク」の動作は、移動コンピュータを含まない状況でも有利に使用することができる。たとえば、この経路指定アルゴリズムは、(リンク状態経路指定アルゴリズムに比べて)少ないメモリ要件を必要とする状況に適用することができる。「アドホック・ネットワーク」の動作は、無線移動コンピュータだけでなく有線移動コンピュータにも適用できる。したがって、一般に、本発明は、新しい宛先順経路指定アルゴリズムを提供し、このアルゴリズムは変30動を弱める技法によって補われる。

【0077】まとめとして、本発明の構成に関して以下の事項を開示する。

【0078】(1)それぞれ固有ネットワーク・アドレスを持つが固定位置を持たない複数の移動ホストから構成されるアドホック・ネットワークに結合された2つの移動ホスト間で、パケットの情報を経路指定する方法であって、各移動ホストにおいて、ソース移動ホストから宛先移動ホストまでのホップ数として定義された「距離」を含む経路指定テーブルを記憶する段階と、各移動ホストに記憶された経路指定テーブルを、その移動ホストが周期的に同報通信することにより経路を公示する段階と、宛先移動ホストから発したタイムスタンプで、各経路指定テーブルのエントリをタグ付けする段階と、他の移動ホストから受け取った同報通信に基づいて、各宛先移動ホストごとに、移動ホストに記憶された経路指定テーブルを更新する段階と、各移動ホストが、隣接移動ホストから受け取った新しい経路指定情報を再伝送する段階と、所望の宛先移動ホストに関する最良の「距離」を有する経路として、ソース移動ホストからパケットの

34

情報を伝送するための経路を選択する段階とを含む方法。

(2)アドホック・ネットワークが、ネットワーク層とリンク層とを含むネットワーク規格に従い、経路指定がアドホック・ネットワークのリンク層で実行されることを特徴とする、上記(1)に記載の経路指定方法。

(3)新しい経路が、より短い距離または無限の距離を有する経路と定義され、無限の距離は、破壊されたリンクを表し、すなわち特定の宛先が到達可能でなくなり、したがってこの新規の到達不可能な宛先に依存する他のすべての宛先がそれ自体到達不可能になることを意味し、隣接移動ホストから受け取った新しい経路指定情報を再伝送する前記段階が、新しい経路指定情報の受信時に移動ホストによって即時実行されることを特徴とする、上記(1)に記載の経路指定方法。

(4)前記経路指定テーブルに記憶された経路が変化する頻度に関するデータを保持する段階と、前記経路指定テーブルに記憶された経路の平均安定待ち時間を決定することによって、経路の安定性を測定する段階と、測定された経路の安定待ち時間を安定待ち時間テーブルに、記憶する段階と、公示段階の前に安定待ち時間テーブルにアクセスし、まもなく変化する可能性のある経路の公示を遅らせ、それにより前記経路指定テーブル内の情報の変動を弱める段階とをさらに含む、上記(1)に記載の経路指定方法。

(5)古い測定値よりも大きな加重因子を有する特定の経路の安定待ち時間の最新の測定値をカウントすることにより、測定された経路安定待ち時間に重み付けを行う段階とをさらに含む、上記(4)に記載の経路指定方法。

(6)アドホック・ネットワークがさらに、移動ホストのネットワーク層アドレスに基づいて、前記経路指定テーブルに経路指定情報を記憶する段階と、前記経路指定テーブルに記憶された経路が変化する頻度に関するデータを保持する段階と、前記経路指定テーブルに記憶された経路の平均安定待ち時間を決定することにより、経路の安定性を測定する段階と、測定された経路の安定待ち

時間を安定待ち時間テーブルに記憶する段階と、公示段階の前に安定待ち時間テーブルにアクセスし、まもなく変化する可能性のある経路の公示を遅らせ、それにより前記経路指定テーブル内の情報の変動を弱める段階とをさらに含む、上記(1)に記載の経路指定方法。

(7) 移動ホストのリンク層アドレスに基づいて、前記経路指定テーブルに経路指定情報を記憶する段階と、前記経路指定テーブルに記憶された経路が変化する頻度に関するデータを保持する段階と、前記経路指定テーブルに記憶された経路の平均安定待ち時間を決定することにより、経路の安定性を測定する段階と、測定された経路の安定待ち時間を安定待ち時間テーブルに記憶する段階と、公示段階の前に安定待ち時間テーブルにアクセスし、まもなく変化する可能性のある経路の公示を遅らせ、それにより前記経路指定テーブル内の情報の変動を弱める段階とをさらに含む、上記(1)に記載の経路指定方法。

(8) ネットワーク層プロトコル使用可能度データを宛先ごとに追跡する段階をさらに含む、上記(1)に記載の経路指定方法。

(9) それぞれ固有ネットワーク・アドレスを持つが固定位置をもたない複数の移動ホストから構成され、ネットワーク層とリンク層とを含み、ネットワーク規格に従うアドホック・ネットワークに結合された2つの移動ホスト間で、パケットの情報を経路指定する方法であって、各移動ホストにおいて、ソース移動ホストから宛先移動ホストまでのホップ数として定義された「距離」を含む経路指定テーブルを記憶する段階と、各移動ホストに記憶された経路指定テーブルを、その移動ホストが周期的に同報通信することにより経路を公示する段階と、宛先移動ホストから発したタイムスタンプで、各経路指定テーブルのエントリをタグ付けする段階と、他の移動ホストから受け取った同報通信に基づき、移動ホストに記憶された経路指定テーブルを移動ホストごとに更新する段階であって、前記更新が、より良い距離または無限の距離を有する経路と定義される新しい経路に限定され、無限の距離が破壊リンクを表し、すなわち、特定の宛先が到達可能でなくなり、したがってこの新規の到達不可能な宛先に依存する他のすべての宛先がそれ自体到達不可能になることを意味する更新段階と、隣接移動ホストから受け取った新しい経路指定情報を各移動ホストが再伝送する段階であって、移動ホストが新しい経路指定情報を受け取ると即時に実行される再伝送段階と、所望の宛先移動ホストに関する最短の「距離」を有する経路として、ソース移動ホストからパケットの情報を伝送する経路を選択する段階とを含む方法。

(10) 前記経路指定テーブルに記憶された経路が変化

する頻度に関するデータを保持する段階と、前記経路指定テーブルに記憶された経路の平均安定待ち時間を決定することにより、経路の安定性を測定する段階と、測定した経路の安定待ち時間を安定待ち時間テーブルに記憶する段階と、公示段階の前に安定待ち時間テーブルにアクセスし、まもなく変化する可能性のある経路の公示を遅らせ、それにより前記経路指定テーブル内の情報の変動を弱める段階とを含む、上記(9)に記載の経路指定方法。

10 (11) 古い事象よりも大きな加重因子を有する特定経路の安定待ち時間の最新の測定値をカウントすることによって、測定された経路安定待ち時間に重み付けを行う段階をさらに含む、上記(10)に記載の経路指定方法。

【発明の効果】変化する任意の相互接続経路に沿って任意の時点でデータを交換できる移動コンピュータ群が、そのコンピュータに、データを変換できる経路を与えることができる、データ通信システムを提供できる。

【図面の簡単な説明】

20 【図1】国防データ・ネットワークのアーキテクチャ図である。

【図2】複数の無線移動ホストからなる「アドホック」ネットワークの機能ブロック図である。

【図3】共通の宛先に接続された、2つの別々の移動ホスト群を示す機能ブロック図である。

【図4】タイムアウト手順の論理を示す流れ図である。

【図5】"ADVERTISE"事象処理の論理を示す流れ図である。

【図6】増分式更新伝送の論理を示す流れ図である。

30 【図7】全ダンプ伝送の論理を示す流れ図である。

【図8】受信時の全ダンプ処理の論理を示す流れ図である。

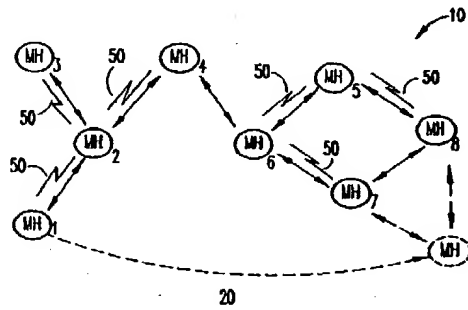
【図9】受信時の増分式更新処理の論理を示す流れ図である。

【図10】安定待ち時間の経過後に、公示経路に経路を挿入する論理を示す流れ図である。

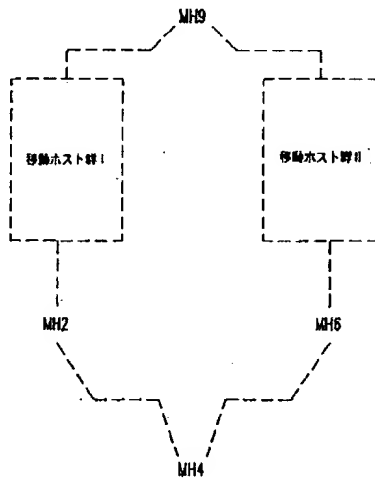
【符号の説明】

- 1 移動ホスト
- 2 移動ホスト
- 40 3 移動ホスト
- 4 移動ホスト
- 5 移動ホスト
- 6 移動ホスト
- 7 移動ホスト
- 8 移動ホスト
- 10 アドホック・ネットワーク
- 50 双方向無線リンク

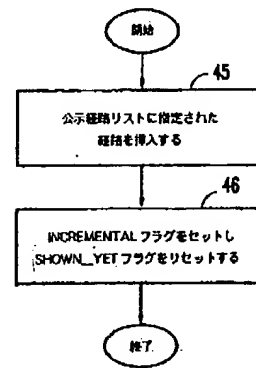
【図2】



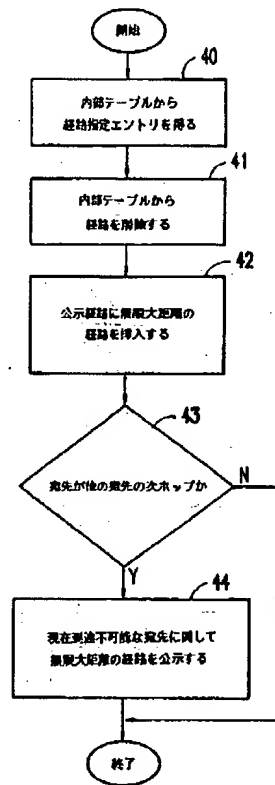
【図3】



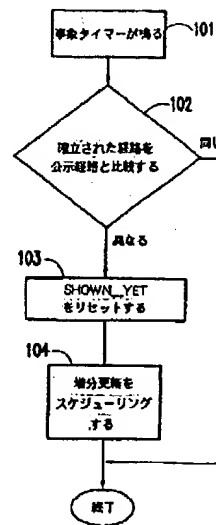
【図5】



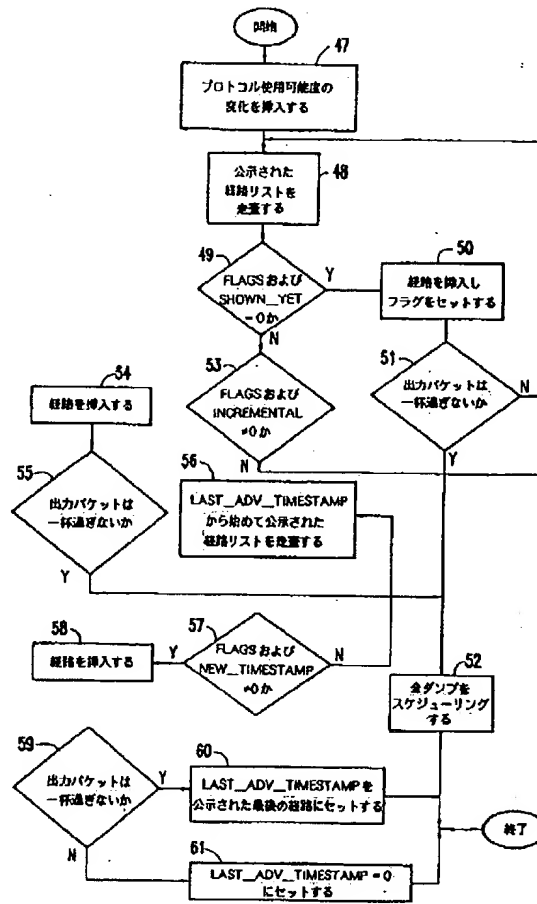
【図4】



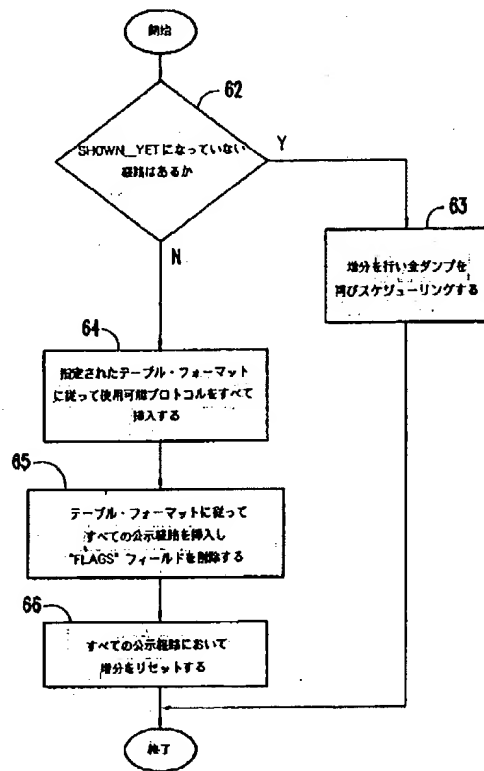
【図10】



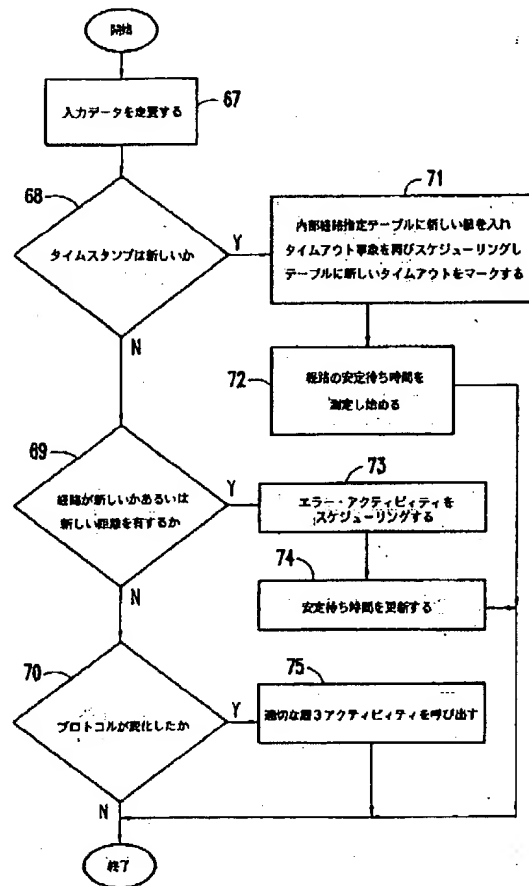
【図6】



【図7】



【図8】



【図9】

